FILE COPY

AD A 121571

TECHNICAL REPORT ARBRL-TR-02439

A DATA BASED RANDOM NUMBER GENERATOR FOR A MULTIVARIATE DISTRIBUTION - A USERS' MANUAL

Barry A. Bodt
Malcolm S. Taylor

November 1982

US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER
TECHNICAL REPORT ARBRL-TR-02439

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
A DATA BASED RANDOM NUMBER GENERATOR FOR A MULTI-VARIATE DISTRIBUTION - A USERS' MANUAL

5. TYPE OF REPORT & PERIOD COVERED

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
Barry A. Bodt
Malcolm S. Taylor

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
US Army Ballistic Research Laboratory
ATTN: DRDAR-BLB
Aberdeen Proving Ground, MD 21005

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
RDT&E 1L161102AH43

11. CONTROLLING OFFICE NAME AND ADDRESS
US Army Armament Research & Development Command
US Army Ballistic Research Laboratory (DRDAR-BL)
Aberdeen Proving Ground, MD 21005

12. REPORT DATE
November 1982

13. NUMBER OF PAGES
32

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report)
UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Density Estimation
Random Number Generator
Nonparametric Statistics
Monte Carlo Simulation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
Let X be a k-dimensional random variable serving as input for a system with output Y (not necessarily of dimension k). Given X, an outcome Y or a distribution of outcomes G(Y|X) may be obtained either explicitly or implicitly. We consider here the situation in which we have a real world data set $\{X_j\}_{j=1}^{n}$ and a means of simulating an outcome Y. A method for empirical random number generation based on the sample of observations of the random variable X without estimating the underlying density is discussed.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

I. INTRODUCTION

An axiom of many simulation studies is that an outcome Y, or distribution of outcomes $G(Y|X)$, of interest can be computer generated using as input experimentally derived data $\{X_j\}_{j=1}^{n}$.

A commonly encountered procedure is one in which a set of experimental data is considered to be a random sample from some underlying but unknown distribution; this data is then modeled by a common statistical distribution to provide a convenient representation (with a coincident loss of information), and is then used as the basis for generating additional pseudo-observations (Monte Carlo values). The intent inherent in this procedure is that the pseudo-observations maintain the statistical structure of the original data set.

The intermediate step of modeling or "fitting" the data as a statistical distribution is sometimes unnecessary and sometimes nearly impossible. For example, for multimodal data or multivariate data, it is usually difficult and often unrealistic to attempt to characterize the data analytically. Because of this fact, there exists little, if any, guidance for the practitioner who is confronted with data of this type. Notice, however, that to serve as input for simulation, all that may actually be required is to provide pseudo-observations that exhibit the same statistical characteristics as the original data set, with no real necessity to formally characterize the underlying distribution.

It is in response to this observation that the following research was initiated and algorithm developed.

II. THE ALGORITHM

Let us consider the following situation addressed by Thompson and Taylor:[1] We have a random sample $\{X_j\}_{j=1}^{n}$ of size n from a multivariate distribution of dimension k, and we want to generate pseudorandom vectors from the underlying, but unknown, distribution that gave rise to the random sample. Since we do not know, and usually will never know, the form of this distribution, our attack should be empirical. We shall endeavor to see to it that our pseudorandom vectors look very much like those in the original data set. In so doing, we will maintain the essential structural integrity of the problem.

We now direct our attention to the mechanics of the algorithm. After carrying out a rough rescaling to account for differing variances that may exist among the k variates, we select at random one of the n data points, say X_1, from the data base and then proceed to determine its m-1 nearest neighbors. The nearest neighbors are determined under the ordinary Euclidean

---

[1] J.R. Thompson and M.S. Taylor, "A Data Based Random Number Generator for a Multivariate Distribution," Proceedings of the Twenty-Seventh Conference on the Design of Experiments in Army Research, Development, and Testing (1981).

metric. The value of m, which can best be determined after perusal of the data, will depend upon the sample size n and the characteristics of the data. A conservative estimate would be to choose $m = n/20$.

The vectors $\{X_j\}_{j=1}^{m}$ are now coded about the sample mean $\bar{X} = 1/m\ \Sigma\ X_i$ to yield $\{X_j'\} = \{X_j - \bar{X}\}_{j=1}^{m}$, and an independent random sample of size m is generated from the uniform distribution $U(1/m - \sqrt{\frac{3(m-1)}{m^2}},\ 1/m + \sqrt{\frac{3(m-1)}{m^2}})$.

Now the linear combination

$$X' = \sum_{\ell=1}^{m} u_\ell X_\ell'$$

is formed, where $\{u_\ell\}_{\ell=1}^{m}$ is the random sample from the $U(1/m - \sqrt{\cdot},\ 1/m + \sqrt{\cdot}\ )$. Finally the translation

$$X = X' + \bar{X}$$

restores the relative magnitude, and X is a pseudorandom vector which we propose to be representative of the multivariate distribution that provided the $\{X_j\}_{j=1}^{n}$.

To obtain the next pseudorandom vector we randomly select another of the n data points and proceed as above.

We will now attempt to advance the algorithm by considering the mathematics that suggests the mechanics that we have just outlined. Consider the distribution of X_1 and its m-1 nearest neighbors: $\{(x_{1\ell}, x_{2\ell}, \ldots, x_{k\ell})'\}_{\ell=1}^{m} = \{X_\ell\}_{\ell=1}^{m}$. Let us suppose that this "truncated set" of random observations has mean vector μ and covariance matrix σ. Let $\{u_\ell\}_{\ell=1}^{m}$ be an independent random sample from the uniform distribution $U(1/m - \sqrt{\cdot},\ 1/m + \sqrt{\cdot}\ )$. Then, $E(u_\ell) = 1/m$, $Var(u_\ell) = (m-1)/m^2$, and $Cov(u_i, u_j) = 0$, for $i \neq j$.

Forming the linear combination

$$Z = \sum_{\ell=1}^{m} u_\ell X_\ell$$

we have, for the r^{th} component $z_r = u_1x_{r1} + u_2x_{r2} + \ldots + u_mx_{rm}$, the following relations

$$E(z_r) = m \cdot 1/m \cdot \mu_r = \mu_r,$$

$$Var(z_r) = \sigma_r^2 + (m-1)/m \cdot \mu_r^2,$$

$$Cov(z_r,z_s) = \sigma_{rs} + (m-1)/m \cdot \mu_r\mu_s.$$

Clearly, if the mean vector of X was $\mu = (0,0, \ldots,0)'$, then the mean vector and covariance matrix of Z would be identical to those of X. In the less idealized situation with which we are confronted, the translation to the sample mean of the nearest neighbor cloud should result in the pseudo-observation having very nearly the same mean and covariance structure as that of the (truncated) distribution of the points in the nearest neighbor cloud, a conjecture substantiated in many actual cases that have been considered. For m moderately large, our algorithm essentially samples from n Gaussian distributions with the means and covariance matrices corresponding to those of the n m-nearest-neighbor clouds.

III. EXAMPLES

For a substantial test case, we considered a mixture of three bivariate normal distributions. The first (N_1) has mean vector $\begin{pmatrix}-1\\-2\end{pmatrix}$ and covariance matrix $\begin{pmatrix}1 & -1/2\\-1/2 & 1\end{pmatrix}$; the second ($N_2$) has mean vector $\begin{pmatrix}-2\\3\end{pmatrix}$ and covariance matrix $\begin{pmatrix}1 & 1/2\\1/2 & 1\end{pmatrix}$; and the third ($N_3$) has mean vector $\begin{pmatrix}2\\3/2\end{pmatrix}$ and covariance matrix $\begin{pmatrix}1 & 1/10\\1/10 & 1\end{pmatrix}$. The corresponding mixing scalars are $\alpha_1 = 1/2$, $\alpha_2 = 1/3$, and $\alpha_3 = 1/6$, respectively. To establish a data base, a sample of eighty-five points was generated from this distribution via Monte Carlo simulation, and appears in Figure 1; a sample of eighty-five pseudorandom values was then produced by the algorithm, and the combined sample is shown in Figure 2.

Notice that the structure of the data is maintained in that the modes are preserved; the algorithm has not attempted to fill in gaps where gaps belong; the algorithm has, however, generated some points outside the boundary of the convex hull of the data base, all of which are desirable properties. These observations lend credence to the term "structural integrity" mentioned previously.

An application of the algorithm to a real world data set is summarized in Figures 3 and 4. In Figure 3, a two-dimensional marginal of a set of 973 four-dimensional behind armor debris measurements is portrayed; in Figure 4 973 simulated data points are produced by our procedure. Once again, the salient features of the data set are preserved.

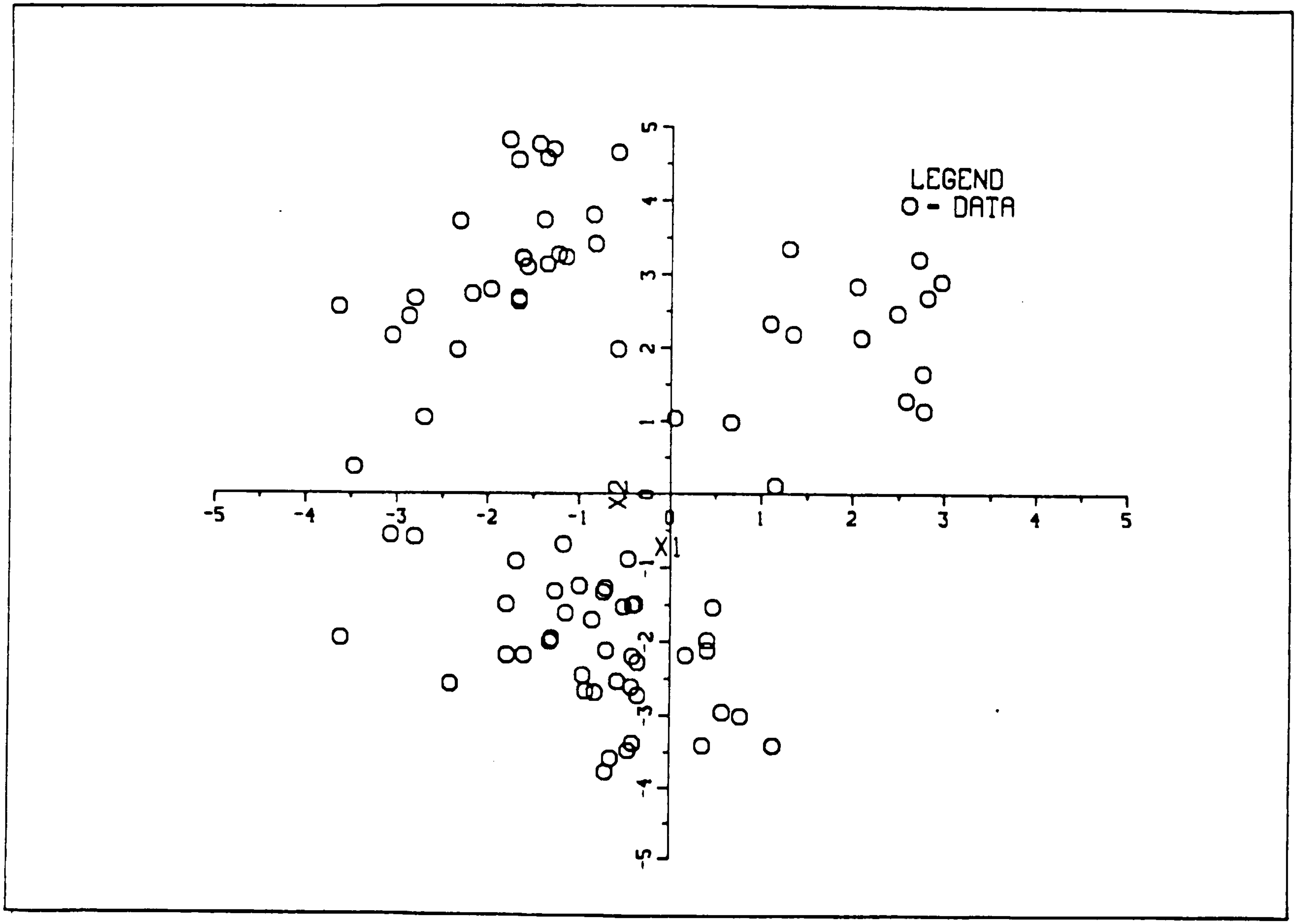

Figure 1. Data base for a mixture of three bivariate normal distributions.

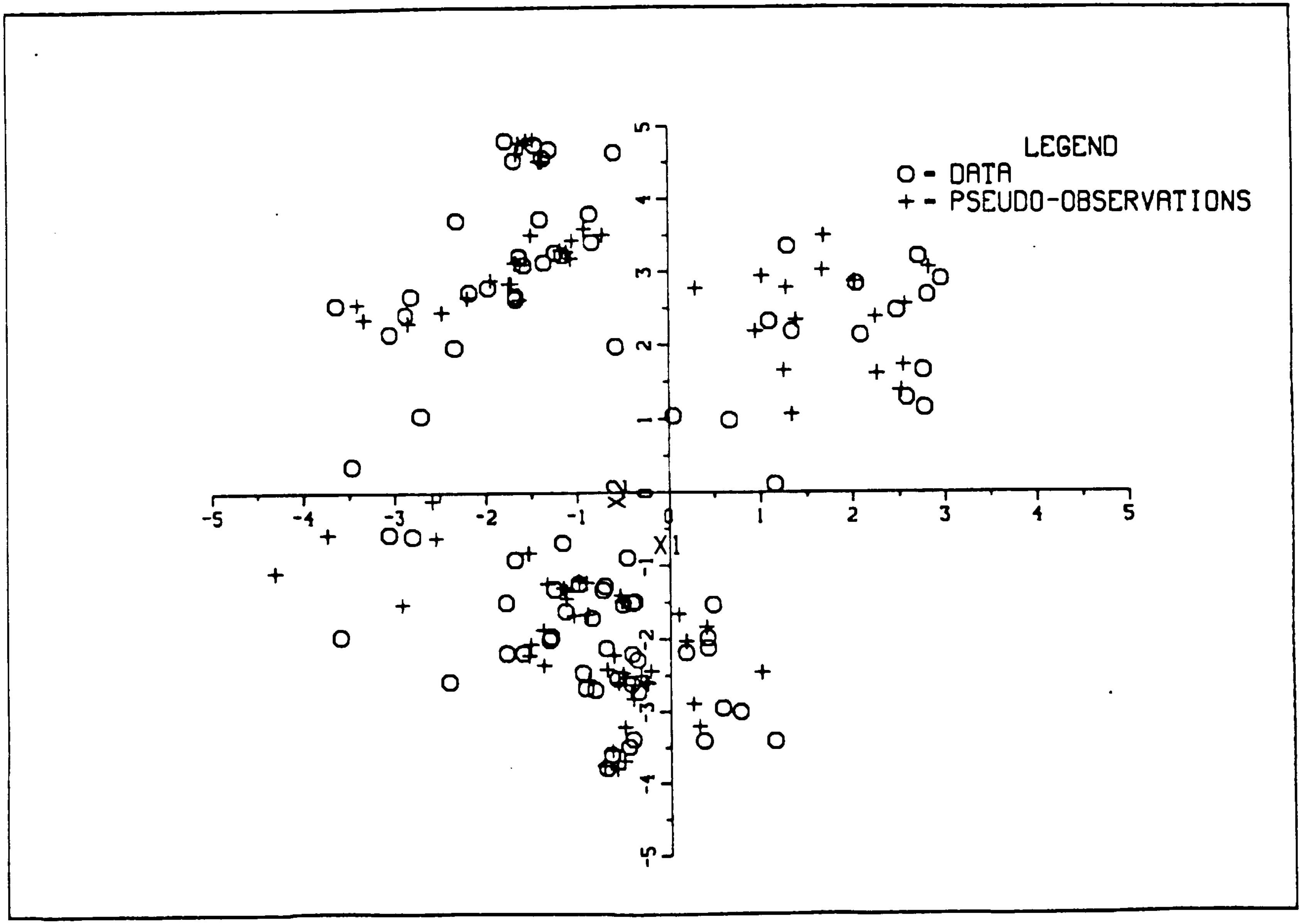

Figure 2. Combined sample: Data base and pseudo-observations.

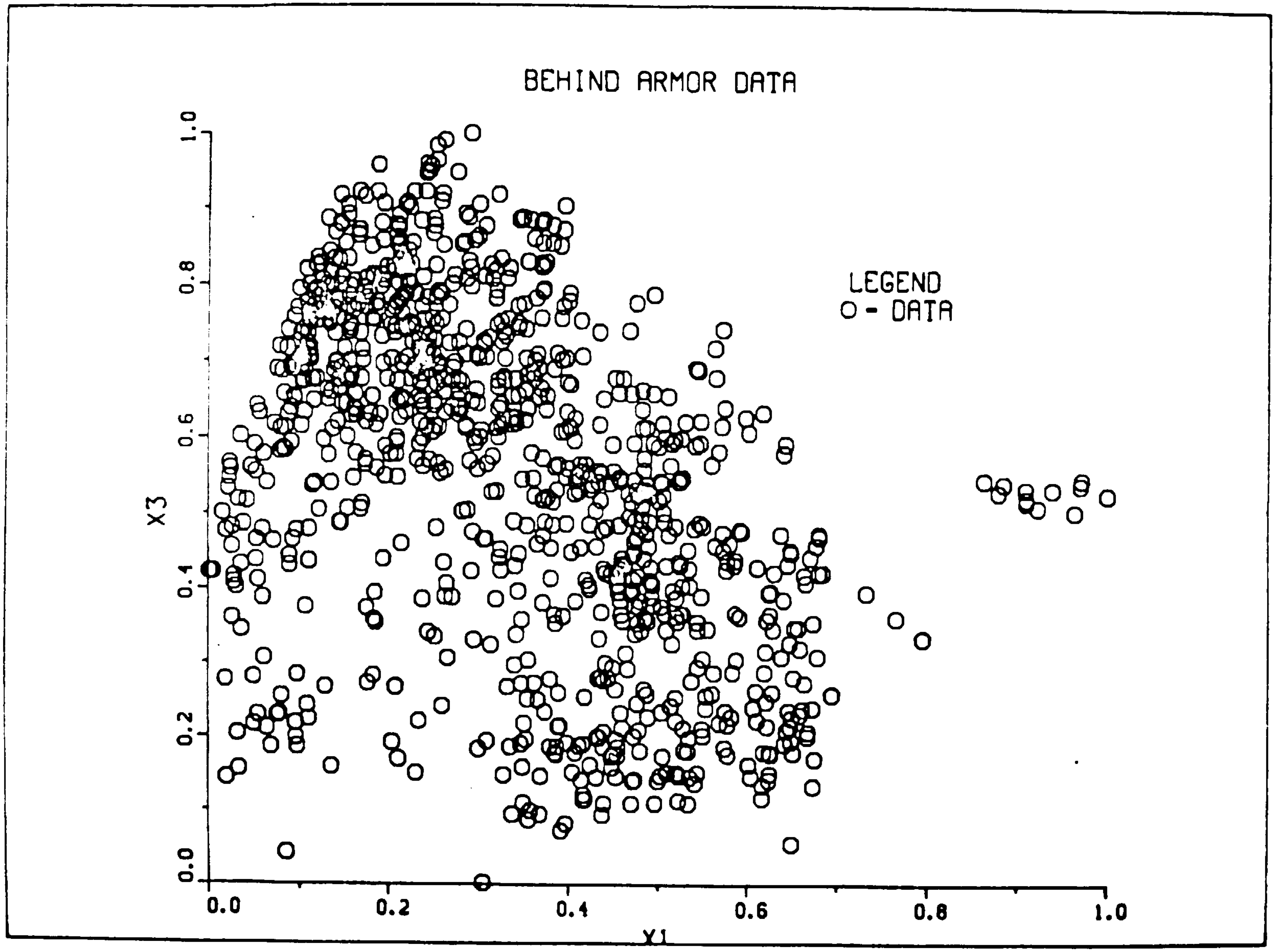

Figure 3. Marginal data for four-dimensional measurements.

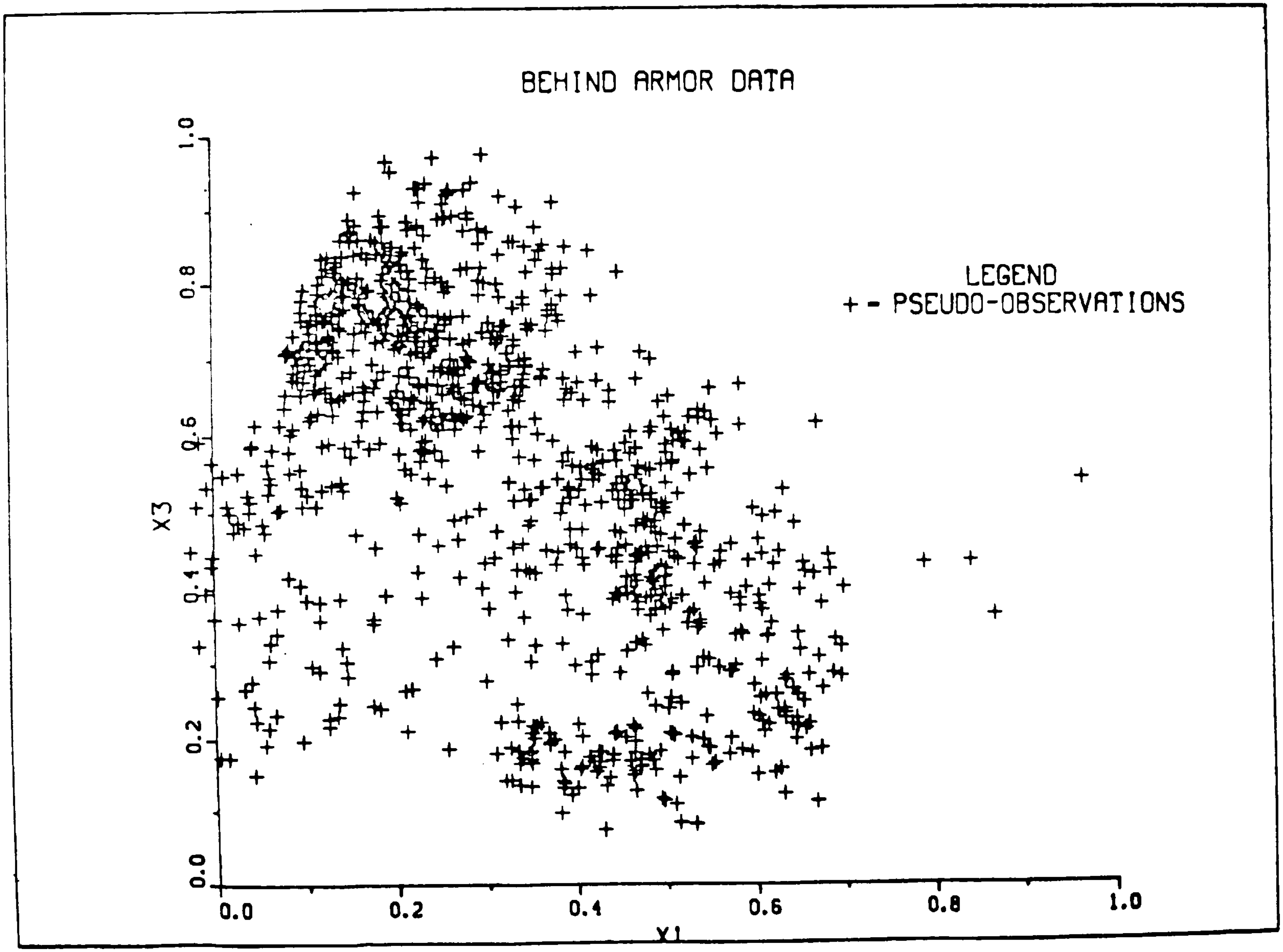

Figure 4. Simulated behind armor debris.

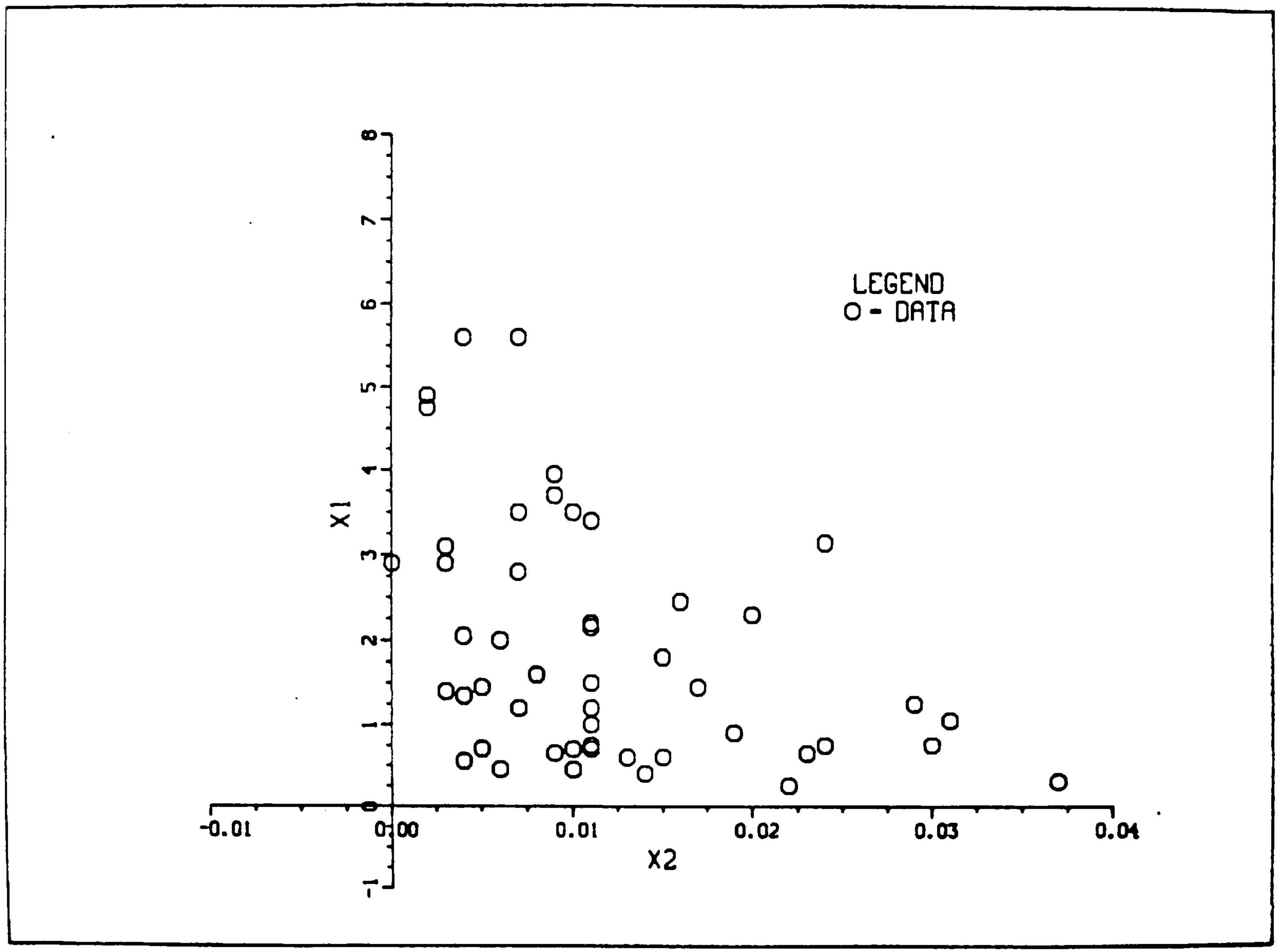

Figure 5. Data base for MLRS bomblets.

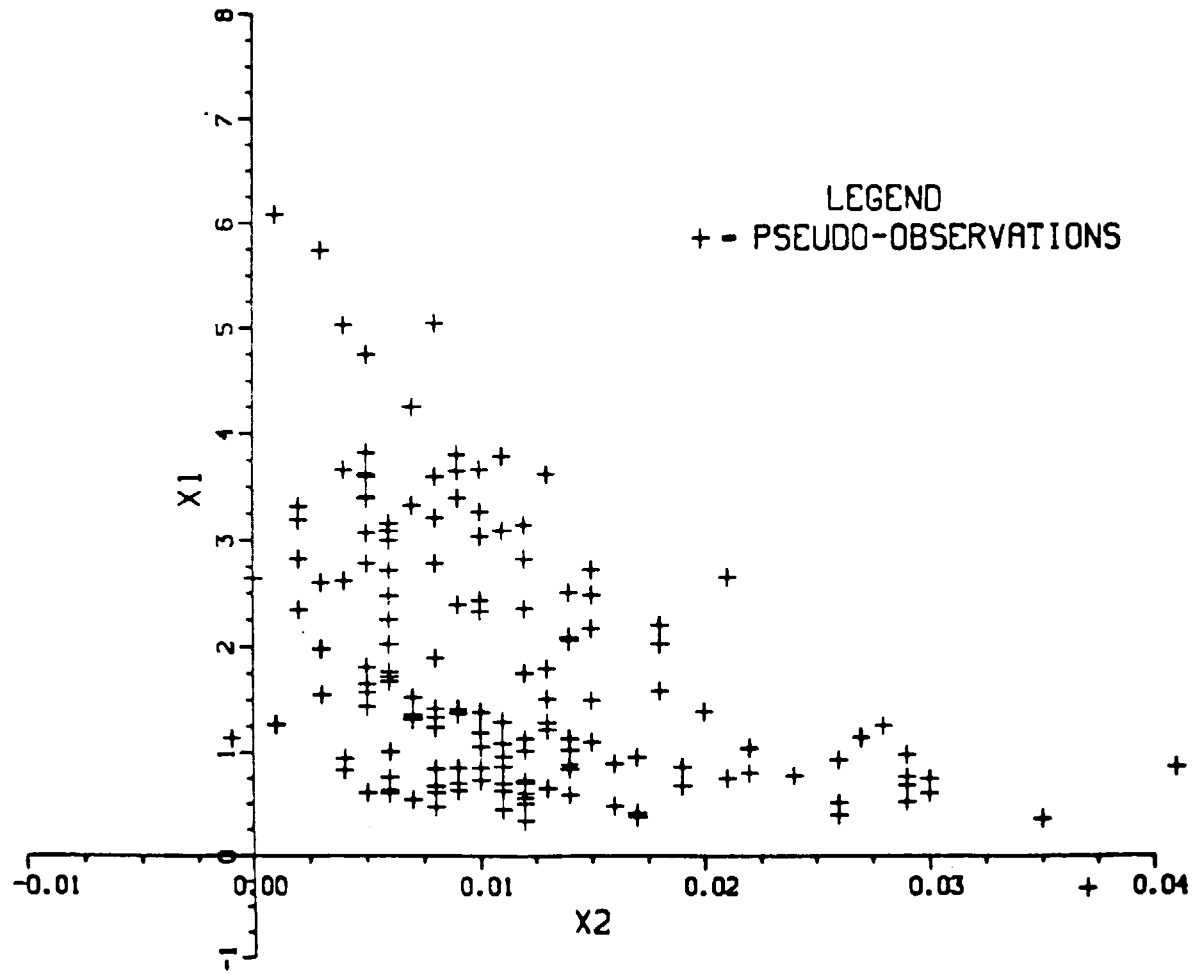

Figure 6. Simulated MLRS bomblet data.

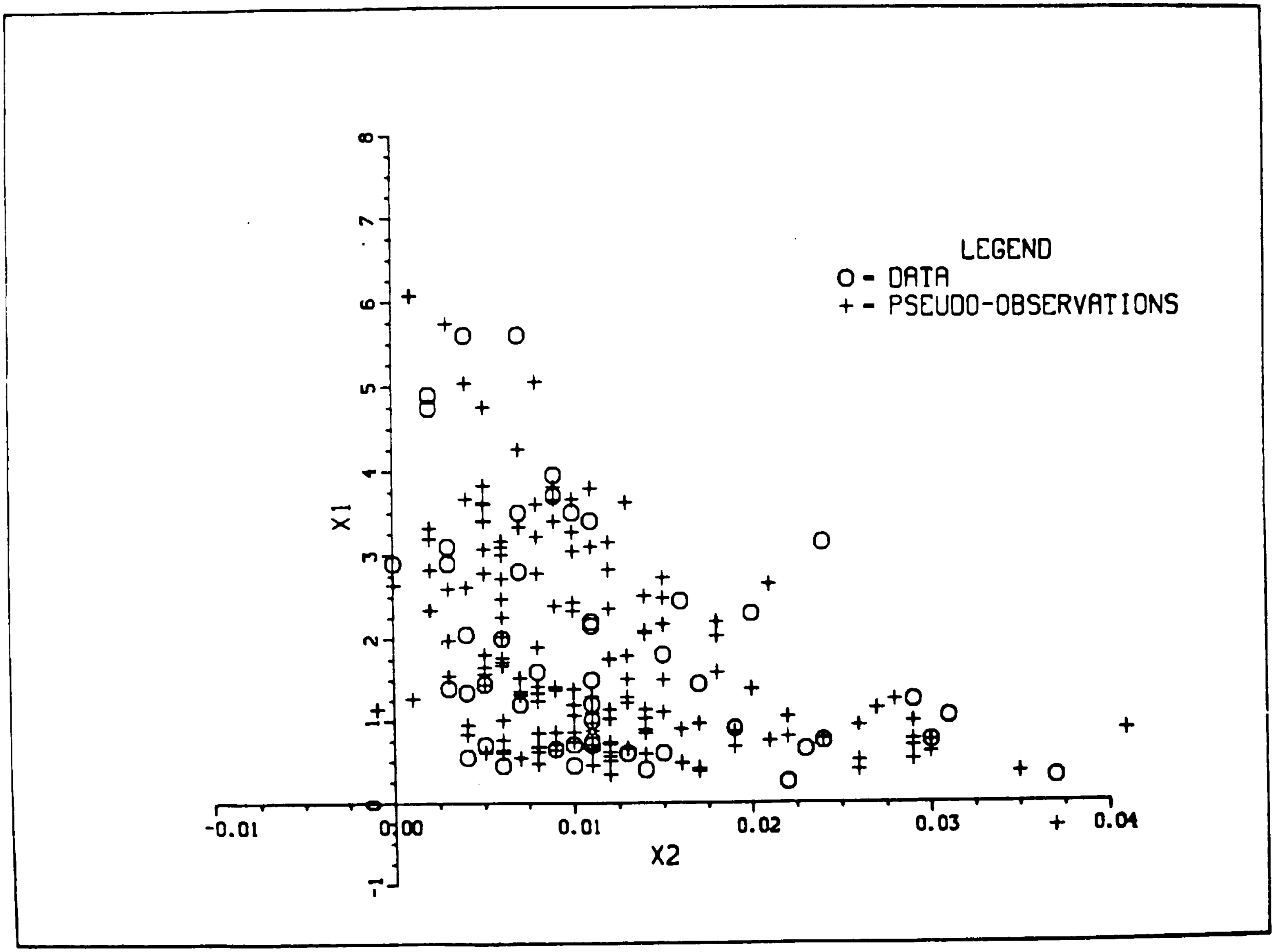

Figure 7. Combined sample: Data base and pseudo-observations.

Figures 5 through 7 display the results of applying the algorithm to a data base of modest size. Here a set of 49 bivariate measurements on MLRS bomblets shown in Figure 5 was supplemented by an additional 160 pseudo-observations (Figure 6), with the results portrayed in Figure 7.

The FORTRAN program of the algorithm appears in Appendix A. This program as listed produces plots using the IMSL Library; Figures 1-7 are plots produced by DISSPLA.

IV. CONCLUSIONS

We have demonstrated a means of empirical random number generation based on a sample of observations of a random variable X. No estimation of the underlying density is required. And, because of the local nature of the generation scheme, it is essentially free of assumptions on the underlying density of X. Naturally, any attempt to use this algorithm for generating bona fide new observations using the computer rather than producing real world data would be unwise. Rather, the algorithm operates somewhat like a smooth interpolator--- highly dependent on the quality of the data points on which it is based. It gives us a means of avoiding nonrobust conclusions due to "holes" in the data set at important points of the simulation model.

ACKNOWLEDGEMENTS

The authors wish to express their appreciation to Csaba K. Zoltani for bringing this problem to their attention and to William E. Baker for suggestions and assistance provided during its pursuit.

APPENDIX A. COMPUTER PROGRAM

STEP 1. CREATION OF THE DATA FILE

Data should be created and stored as an MFA permanent file. This file should contain the following information. Note: Let pl denote program limitations.

CARD 1. 5 INPUTS	FORMAT	COLUMNS
1. NUMDAT - number of input data points	I4	1-4
pl. $5 \leq$ NUMDAT ≤ 1000		
2. NUMGEN - number of pseudopoints generated	I4	6-9
pl. $5 \leq$ NUMGEN ≤ 1000		
3. IDIM - dimension n of n-space data	I1	11
pl. $1 \leq$ IDIM ≤ 8		
4. NRANDM - random number seed	I3	16-18
pl. $1 \leq$ NRANDM ≤ 999		
5. NUMPLT - number of plots requested	I2	21-22

Note - all possible 2-d plots (NUMPLT = -1)
- no plots (NUMPLT = 0)
- r plots (NUMPLT = r)
- NUMPLT plots will be generated for both data and pseudo-observations.

pl. $-1 \leq$ NUMPLT $\leq$ IDIM(IDIM-1)/2

CARD 2 & CARD 3

If NUMPLT = r, request plots of variables X_i vs. Y_j by indicating j in Card 2 and the corresponding i in Card 3.

Column	1	2	...	r
Card(2)	Y_1	Y_2	...	Y_r
Card(3)	X_1	X_2	...	X_r

If NUMPLT = 0 or -1, cards 2 and 3 should not be included, otherwise information on these cards would be used as data.

Card(4), Card(5) ... Card(NUMDAT)

These cards should contain data to be read in with F10.3 format. Data may consist of a maximum of 8 variables X_1 - X_8.

Columns	1 - 10	11 - 20	...	71 - 80
Card(4)	X_1 Data	X_2 Data		X_8 Data
Card(5)	X_1 Data	X_2 Data		X_8 Data
.	.	.		.
.	.	.		.
.	.	.		.
Card(NUMDAT)	X_1 Data	X_2 Data		X_8 Data

STEP 2

Before the program can be run, the data file must be made accessible to the MFZ with (PERMIT, PFN, MFZ).

STEP 3

To run the program create and submit the following 3 card MFZ job.

JOBNAME, STMFZ, T100.

ACCOUNT, XXXXXXX.

BEGIN, DBRNG, DBRNG, PFI = _____, PFO = _____, UN = _____, RJE = RJEXXXX.

Where

PFI = file name under which the input data file is stored,

PFO = file name under which the pseudo data is to be stored,

UN = user name identification for above two files,

XXXX = a 4 digit code designating a particular RJE, as the output device. If omitted, then the central site will serve as the output destination.

```
C THIS PROGRAM PRODUCES PSEUDORANDOM OBSERVATIONS FROM REAL
C DATA FOR UNIVARIATE AND MULTIVARIATE CASES.  THE PSEUDO-
C RANDOM OBSERVATIONS WILL MAINTAIN THE CHARACTERISTICS OF
C THE REAL DATA WITHOUT ANY DISTRIBUTION ASSUMPTION ON THE
C POPULATION FROM WHICH THE REAL DATA CAME.  AN EXAMPLE OF
C PROPER USE OF THIS PROGRAM WOULD BE IN THE CREATION OF PSEUDO-
C RANDOM OBSERVATIONS FOR INPUT TO A COMPUTER SIMULATION
C MODEL.
C
C CAUTION! THIS PROGRAM DOES NOT PRODUCE REAL DATA AND PRODUCED
C          PSEUDORANDOM OBSERVATIONS SHOULD NOT BE USED AS SUCH.
C
C NUMDAT- NUMBER OF INPUT DATA POINTS
C NUMGEN- NUMBER OF PSEUDOPOINTS TO BE GENERATED
C IDIM-   NUMBER OF VARIABLES IN INPUT DATA SET
C NRANDM- RANDOM NUMBER SEED
C NUMPLT- NUMBER OF PLOTS REQUESTED
C DATA-   MATRIX HOLDING INPUT DATA SET
C NPLT-   MATRIX HOLDING PLOT REQUESTS
C
      PROGRAM DBRNG(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE9)
C ------------------------------------------------------------
      DIMENSION DATA(1000,10),PSEUDO(1000,10),NPLT(2,40),ZMIN(10)
      DIMENSION ZMAX(10)
C
C READ AND WRITE INITIAL INPUT VARIABLES
C
      READ(8,1000)NUMDAT,NUMGEN,IDIM,NRANDM,NUMPLT
      WRITE(6,2030)
      WRITE(6,2040)NUMDAT,NUMGEN,IDIM,NRANDM,NUMPLT
C
C CHECK FOR INVALID VALUES OF INITIAL INPUT VARIABLES
C
      CALL CHECK(NUMDAT,NUMGEN,IDIM,NRANDM,NUMPLT,NCHECK)
      IF(NCHECK.EQ.1)GO TO 90
      IF(NUMPLT.GT.0)GO TO 10
C
C ESTABLISH NPLT FOR ALL POSSIBLE IDIM(IDIM-1)/2 PLOTS
C
      CALL SETPLT(NPLT,IDIM)
      GO TO 40
C
C USER ESTABLISHES DESIRED PLOTS
C Y VALUES FIRST CARD, CORRESPONDING X VALUES SECOND CARD
C
   10 DO 20 K=1,2
      READ(8,1010)(NPLT(K,L),L=1,NUMPLT)
   20 CONTINUE
C
C IF USER DEFINED, WRITE PLOTS REQUESTED
C
      WRITE(6,2050)
      DO 30 K=1,2
      WRITE(6,2070)(NPLT(K,L),L=1,NUMPLT)
   30 CONTINUE
C
C READ REAL DATA AND WRITE FIRST FIVE POINTS
C
   40 DO 50 I=1,NUMDAT
      READ(8,1020)(DATA(I,J),J=1,IDIM)
   50 CONTINUE
      WRITE(6,2060)
      DO 60 I=1,5
      WRITE(6,2020)(DATA(I,J),J=1,IDIM)


   60 CONTINUE
```

```
C VARY THE RANDOM NUMBER SEQUENCE USING INPUT VARIABLE NRANDM
C
      DO 70 K=1,NRANDM
      RN=RANF(0)
   70 CONTINUE
C
C WRITE HEADER FOR OUTPUT
C
      WRITE(6,2000)
C
C DETERMINE THE CORRELATION MATRIX, MEAN AND VARIANCES FOR REAL DATA
C
      CALL CORREL(DATA,IDIM,NUMDAT)
C
C SCALE DATA SO THAT EACH VARIABLE WILL CARRY EQUAL WEIGHT IN
C THE NEIGHBORHOOD SELECTION PROCESS
C
      CALL SCALE(DATA,NUMDAT,IDIM,ZMIN,ZMAX)
C
C GENERATE NUMGEN PSEUDODATA POINTS
C
      CALL GNERAT(DATA,NUMDAT,NUMGEN,IDIM,PSEUDO)
C
C RESCALE THE DATA AND THE CORRESPONDING PSEUDODATA TO THEIR
C ORIGINAL MAGNITUDES
C
      CALL RESCAL(DATA,NUMDAT,IDIM,ZMIN,ZMAX)
      CALL RESCAL(PSEUDO,NUMGEN,IDIM,ZMIN,ZMAX)
C
C WRITE HEADER FOR OUTPUT
C
      WRITE(6,2010)
C
C DETERMINE THE CORRELATION MATRIX, MEAN AND VARIANCES FOR PSEUDO-
C DATA
C
      CALL CORREL(PSEUDO,IDIM,NUMGEN)
C
C WRITE THE PSEUDORANDOM OBSERVATIONS ONTO A PERMANENT
C FILE ( PSEUDO )
C
      DO 80 J=1,NUMGEN
      WRITE(9,2020)(PSEUDO(J,L),L=1,IDIM)
   80 CONTINUE
C
C CALL PLOT ROUTINE IF REQUESTED.
C
      IDUMY=IDIM*(IDIM-1)/2
      IF(NUMPLT.EQ.-1)NUMPLT=IDUMY
      IF(NUMPLT.LE.0)GO TO 90
      CALL PLOT(PSEUDO,DATA,ZMIN,ZMAX,NPLT,NUMDAT,NUMGEN,NUMPLT)
   90 CONTINUE
      WRITE(6,2080)
      STOP
 1000 FORMAT(I4,1X,I4,1X,I1,4X,I3,2X,I2)
 1010 FORMAT(36I1)
 1020 FORMAT(8F10.3)
 2000 FORMAT(1H1,20X,'CORRELATIONS, MEANS AND VARIANCES OF INPUT DATA SE
     +T',/)
 2010 FORMAT(///,20X,'CORRELATIONS, MEANS AND VARIANCES OF PSEUDO DATA S
     +ET',/)
 2020 FORMAT(4X,8(F12.3,3X))


 2030 FORMAT(1H1,4X,'NUMDAT',2X,'NUMGEN',3X,'IDIM',3X,'NRANDM',
     +2X,'NUMPLT')
 2040 FORMAT(/,5X,5(I4,4X))
 2050 FORMAT(///,5X'PLOTS REQUESTED     Y OVER X',/)
 2060 FORMAT(///,5X,'FIRST FIVE DATA POINTS',/)
 2070 FORMAT(5X,36I2)
 2080 FORMAT(1H1,' ')
      END
```

```
C
C
C
C THIS SUBROUTINE CHECKS THE INITIAL INPUT DATA FOR VALUES
C WHICH WILL CAUSE THE PROGRAM TO FAIL.  FOR EXAMPLE NUMDAT
C CAN BE A MAXIMUM OF 1000 AS THE DIMENSION STATEMENT ONLY
C ALLOWS FOR 1000 DATA POINTS.  IF AN INCORRECT VALUE IS
C DETECTED, NCHECK WILL BE SET TO 1.  WHEN RETURNED AS 1
C THE PROGRAM WILL STOP.  IF RETURNED AS 0 THE PROGRAM WILL
C CONTINUE NORMALLY.
C
      SUBROUTINE CHECK(NUMDAT,NUMGEN,IDIM,NRANDM,NUMPLT,NCHECK)
C ---------------------------------------------------------------
      IF(NUMDAT.GT.1000.OR.NUMDAT.LT.5)GO TO 100
      IF(NUMGEN.LT.5.OR.NUMGEN.GT.1000)GO TO 200
      IF(IDIM.LT.1.OR.IDIM.GT.8)GO TO 300
      IF(NRANDM.LE.0)GO TO 400
      K=IDIM*(IDIM-1)/2
      IF(NUMPLT.GT.K)GO TO 500
      IF(NUMPLT.LT.-1)GO TO 500
      IF(IDIM.EQ.1.AND.NUMPLT.EQ.-1)GO TO 500
      NCHECK=0
      RETURN
  100 WRITE(6,2000)
      NCHECK=1
      RETURN
  200 WRITE(6,2010)
      NCHECK=1
      RETURN
  300 WRITE(6,2020)
      NCHECK=1
      RETURN
  400 WRITE(6,2030)
      NCHECK=1
      RETURN
  500 WRITE(6,2040)
      NCHECK=1
      RETURN
 2000 FORMAT(//,5X,'INVALID NUMBER OF INPUT DATA POINTS')
 2010 FORMAT(//,5X,'INVALID NUMBER OF PSEUDO DATA POINTS')
 2020 FORMAT(//,5X,'INVALID DIMENSION N OF N-SPACE DATA')
 2030 FORMAT(//,5X,'INVALID SEED')
 2040 FORMAT(//,5X,'INVALID NUMBER OF 2D PLOTS FOR DIMENSION SPECIFIED')
      END
C
C
C
C THIS SUBROUTINE INTIALIZES THE NPLT MATRIX SO THAT ALL POSSIBLE
C 2-D PLOT COMBINATIONS ARE CONSIDERED.  THERE IS A TOTAL OF
C IDIM(IDIM-1)/2 PLOTS WHICH COULD BE MADE.  IN MAIN IF NUMPLT=0, NO
C PLOTS WILL BE MADE.  IF NUMPLT=-1, ALL PLOTS WILL BE MADE.
C
      SUBROUTINE SETPLT(NPLT,IDIM)
C -----------------------------------
      DIMENSION NPLT(2,40)
      K=1


      II=IDIM-1
      DO 20 I=1,II
      JJ=I+1
      DO 10 J=JJ,IDIM
      NPLT(1,K)=I
      NPLT(2,K)=J
      K=K+1
   10 CONTINUE
   20 CONTINUE
      RETURN
      END
```

```
C THIS SUBROUTINE SCALES THE DATA SO THAT EACH VARIABLE WILL
C CARRY EQUAL WEIGHT IN THE NEIGHBORHOOD SELECTION PROCESS.  THE
C SCALED DATA WILL THEN BE RETURNED TO MAIN.   THE PROCESS USED
C IS   ( X(I)-MIN(X(I)) ) / RANGE(X(I))     FOR EACH VARIABLE.
C
      SUBROUTINE SCALE(TDATA,NSORT,IDIM,ZMIN,ZMAX)
C ----------------------------------------------------
      DIMENSION TDATA(1000,10),ZMIN(10),ZMAX(10)
C
C INPUT FOR BUBBLE SORT
C
      NTOP=NSORT-1
C
C LOOP WHICH SORTS ON EACH VARIABLE ( NRANK ) AND THEN
C ESTABLISHES ITS MINIMUM AND MAXIMUM ( ZMIN ) AND ( ZMAX )
C RESPECTIVELY
C
      DO 10 I=1,IDIM
      NRANK=I
      CALL SORT(TDATA,NSORT,NTOP,NRANK,IDIM)
      ZMIN(I)=TDATA(1,I)
      ZMAX(I)=TDATA(NSORT,I)
   10 CONTINUE
C
C LOOP WHICH PERFORMS THE ABOVE MENTIONED TRANSFORMATION
C
      DO 30 J=1,NSORT
      DO 20 K=1,IDIM
      TDATA(J,K)=(TDATA(J,K)-ZMIN(K))/(ZMAX(K)-ZMIN(K))
   20 CONTINUE
   30 CONTINUE
      RETURN
      END
C
C
C
C THIS ROUTINE SORTS THE DATA MATRIX ON POSITION NRANK
C THE SORT USED IS A COMMON BUBBLE SORT WHICH WILL ESTABLISH
C THE FIRST NTOP POINTS FROM SMALLEST TO LARGEST - WHERE
C SMALLEST TO LARGEST IS DETERMINED BY POSITION NRANK.  NOTE
C THAT WHEN AN EXCHANGE TAKES PLACE THE ENTIRE ROW VECTOR,
C SOME POINT (W,X,Y,....,D) IS EXCHANGED.  NOTE ALSO THAT D
C REPRESENTS THE DISTANCE SQUARED COMPUTED IN EUCLID AND STORED
C IN POSITION IDIST.
C
C
C
      SUBROUTINE SORT(SDATA,NSORT,NTOP,NRANK,IDIM)
C ----------------------------------------------------
      DIMENSION SDATA(1000,10)


      IDIST=IDIM+1
C
C TAKE THE FIRST I'TH VALUE AND COMPARE IT TO THE I+1'TH
C VALUE.  IF THE I'TH VALUE IS SMALLER, EXCHANGE IT WITH
C THE I+1'TH VALUE SO THAT THE I'TH VALUE IS THEN SMALLER.
C THEN COMPARE THE I'TH VALUE WITH THE I+2'TH VALUE AND
C SO ON.
C
      DO 30 I=1,NTOP
      L=I+1
      DO 20 J=L,NSORT
      IF(SDATA(I,NRANK).LT.SDATA(J,NRANK))GO TO 20
      DO 10 K=1,IDIST
      TEMP=SDATA(I,K)
      SDATA(I,K)=SDATA(J,K)
      SDATA(J,K)=TEMP
   10 CONTINUE
   20 CONTINUE
   30 CONTINUE
      RETURN
      END
```

```
C
C
C
C THIS SUBROUTINE DOES THE ACTUAL GENERATION OF THE PSEUDORANDOM
C OBSERVATIONS, AND RETURNS THEM IN A MATRIX ( PSEUDO ).  THE
C ALGORITHM USED WAS DEVELOPED BY DR. JIM THOMPSON OF RICE UNIVERSITY
C AND DR. MALCOLM TAYLOR OF BRL.
C
      SUBROUTINE GNERAT(DATA,NUMDAT,NUMGEN,IDIM,PSEUDO)
C ------------------------------------------------------------
      DIMENSION DATA(1000,10),PSEUDO(1000,10),AVERAG(10)
      DIMENSION TRANSD(25,10)
C
C INITIALIZE THE MATRIX PSEUDO TO ZERO.
C
      DO 20 L=1,NUMGEN
      DO 10 K=1,IDIM
      PSEUDO(L,K)=0.
   10 CONTINUE
   20 CONTINUE
C
C ESTABLISH THE SIZE OF THE NEIGHBORHOOD OF NEAREST POINTS TO
C BE USED IN A LINEAR COMBINATION.
C
      NEIGHB=INT(FLOAT(NUMDAT)/20.)
      IF(NEIGHB.LT.5)NEIGHB=5
      IF(NEIGHB.GT.20)NEIGHB=20
C
C IDIST MARKS THE COLUMN WHERE THE EUCLIDEAN DISTANCE SQUARED
C WILL BE STORED.
C
      IDIST=IDIM+1
C
C INITIALIZE THOSE DISTANCES AS ZERO TO PREVENT COMPUTER ERROR
C
      DO 30 J=1,NUMDAT
      DATA(J,IDIST)=0.
   30 CONTINUE
C
C WEIGHT IS THE WEIGHTING FACTOR TO BE USED IN CALCULATING THE
C MEAN OF THE NEIGHB NEAREST NEIGHBORS.  IT ALSO SERVES AS THE
C MEAN OF THE SPECIAL UNIFORM DISTRIBUTION USED IN THE LINEAR
C COMBINATION.


C
      WEIGHT=1./FLOAT(NEIGHB)
C
C UNADJ1 HELPS DEFINE THE UNIFORM DISTRIBUTION WITH MEAN
C WEIGHT.
C
      UNADJ1=(3.*(FLOAT(NEIGHB)-1.)/(FLOAT(NEIGHB)**2.))**.5
C
C THE FOLLOWING LOOP GENERATES NUMGEN PSEUDORANDOM
C OBSERVATIONS.
C
      DO 120 JJJ=1,NUMGEN
C
C INITIALIZE THE AVERAG ARRAY EACH TIME A NEW POINT IS CHOSEN.
C
      DO 40 NSET=1,IDIM
      AVERAG(NSET)=0.
   40 CONTINUE
```

```
C
C RANDOMLY PICK A DATA POINT ( KCENTR ) AROUND WHICH A
C A NEIGHBORHOOD WILL BE FORMED.
C
      RN=RANF(0)
      KCENTR=INT(RN*FLOAT(NUMDAT))+1
C
C ESTABLISH THE EUCLIDEAN DISTANCE SQUARED OF ALL POINTS
C FROM KCENTR.
C
      CALL EUCLID(DATA,NUMDAT,IDIM,KCENTR)
C
C SORT THE POINTS ON THEIR EUCLIDEAN DISTANCE FROM
C SMALLEST TO LARGEST.  IN THIS FASHION THE NEIGHB NEAREST
C NEIGHBORS WILL BE CHOSEN.
C
      CALL SORT(DATA,NUMDAT,NEIGHB,IDIST,IDIM)
C
C COMPUTE THE AVERAGE OF X,Y,Z,... IN (X,Y,Z,...) OF KCENTR
C AND ITS NEAREST NEIGHBORS
C
      DO 60 I=1,NEIGHB
      DO 50 J=1,IDIM
      AVERAG(J)=AVERAG(J)+DATA(I,J)*WEIGHT
   50 CONTINUE
   60 CONTINUE
C
C CREATE A TRANSLATED DATA SET ( TRANSD ) TO BE USED IN
C THE CREATION OF ONE POINT.
C
      DO 80 M=1,NEIGHB
      DO 70 L=1,IDIM
      TRANSD(M,L)=DATA(M,L)-AVERAG(L)
   70 CONTINUE
   80 CONTINUE
C
C BEGIN THE LOOP WHICH CREATES A NEW POINT BY TAKING
C A LINEAR COMBINATION OF THE TRANSLATED DATA.
C
      DO 100 I=1,NEIGHB
C
C ESTABLISH A RANDOM NUMBER FROM THE SPECIAL UNIFORM
C DISTRIBUTION TO BE MULTIPLIED BY ONE DATA VECTOR
C (X,Y,Z,...).
C
      RN=RANF(0)


      RN=RN*2*UNADJ1+(WEIGHT-UNADJ1)
C
C LOOP WHICH ADDS THE TRANSFORMED VECTORS TO CREATE
C ONE NEW POINT
C
      DO 90 J=1,IDIM
      PSEUDO(JJJ,J)=PSEUDO(JJJ,J)+RN*TRANSD(I,J)
   90 CONTINUE
  100 CONTINUE
C
C LOOP WHICH ADDS BACK IN THE AVERAGE OF THE NEIGHBORHOOD
C WHICH WAS TAKEN AWAY FROM ( TRANSD )
C
      DO 110 L=1,IDIM
      PSEUDO(JJJ,L)=PSEUDO(JJJ,L)+AVERAG(L)
  110 CONTINUE
  120 CONTINUE
      RETURN
      END
```

```
C
C
C
C THIS SUBROUTINE COMPUTES THE MEAN VECTOR, VARIANCE VECTOR
C AND CORRELATION MATRIX OF ANY MATRIX SUBMITTED TO IT.  IT THEN
C WRITES THIS INFORMATION ON HARD COPY.
C
      SUBROUTINE CORREL(CDATA,IDIM,NUMCOR)
C ---------------------------------------------
      DIMENSION CDATA(1000,10),SUMXI(10),SUMXY(10,10),VAR(10),LINE(8)
      DIMENSION AVR(10),CORR(10,10),LINE1(40),LINE2(18)
C
C INSURES OUTPUT NEATNESS
C
      DATA LINE/'X1','X2','X3','X4','X5','X6','X7','X8'/
      DATA LINE2/'C','O','R','R','E','L','A','T','I','O','N',' ',
     +'M','A','T','R','I','X'/
      DO 10 I=1,40
      LINE1(I)=' '
   10 CONTINUE
C
C INITIALIZE THE SUM OF X(I) AND THE SUM OF X(I)*X(J) TO ZERO.
C
      DO 30 M=1,IDIM
      SUMXI(M)=0.
      DO 20 MM=1,IDIM
      SUMXY(M,MM)=0.
   20 CONTINUE
   30 CONTINUE
C
C COMPUTE THE SUM OF X(I) AND THE SUM OF X(I)*X(J).
C
      DO 60 I=1,NUMCOR
      DO 50 J=1,IDIM
      SUMXI(J)=SUMXI(J)+CDATA(I,J)
      KK=J
      DO 40 K=KK,IDIM
      SUMXY(J,K)=SUMXY(J,K)+CDATA(I,J)*CDATA(I,K)
   40 CONTINUE
   50 CONTINUE
   60 CONTINUE
C
C GET THE REAL VALUE OF THE NUMBER OF POINTS CONSIDERED.
C
      W=NUMCOR


C
C COMPUTE THE MEANS AND SAMPLE VARIANCES OF X,Y,Z... OF
C (X,Y,Z,...).
C
      DO 70 M=1,IDIM
      AVR(M)=SUMXI(M)/W
      VAR(M)=(SUMXY(M,M)/W-AVR(M)**2.)*W/(W-1.)
   70 CONTINUE
C
C COMPUTE THE CORRELATION OF X(L) AND X(K).
C
      DO 90 L=1,IDIM
      KK=L
      DO 80 K=KK,IDIM
      CORR(L,K)=SUMXY(L,K)-SUMXI(L)*SUMXI(K)/W
      CORR(L,K)=CORR(L,K)/((W-1.)*(VAR(L)*VAR(K))**.5)
   80 CONTINUE
   90 CONTINUE
```

```
C
C THE FOLLOWING ARE ALL AIDS IN WRITING THE INFORMATION IN
C AN ACCEPTABLE FASHION.
C
      TEMP=FLOAT(IDIM)
      NTO=INT(TEMP/2.*9.)-5
      WRITE(6,2000)(LINE1(J),J=1,NTO),(LINE2(I),I=1,18)
      WRITE(6,2010)(LINE(K),K=1,IDIM)
      DO 100 I=1,IDIM
      IF(I.EQ.IDIM)GO TO 110
      K=I+1
      WRITE(6,2020)I,(CORR(J,I),J=1,I),(CORR(I,J),J=K,IDIM)
  100 CONTINUE
  110 WRITE(6,2020)IDIM,(CORR(J,IDIM),J=1,IDIM)
      WRITE(6,2030)(LINE(K),K=1,IDIM)
      WRITE(6,2040)(AVR(I),I=1,IDIM)
      WRITE(6,2050)(VAR(J),J=1,IDIM)
 2000 FORMAT(1H0,58A1)
 2010 FORMAT(/,9X,A2,7(7X,A2))
 2020 FORMAT(//,1X,'X',I1,3X,8(F8.5,1X))
 2030 FORMAT(///,7X,8(10X,A2))
 2040 FORMAT(/,2X,'MEAN',3X,8(1X,F11.3))
 2050 FORMAT(/,2X,'VAR',4X,8(1X,F11.3))
      RETURN
      END
C
C
C
C THIS SUBROUTINE CALCULATES THE EUCLIDEAN DISTANCE
C SQUARED BETWEEN KCENTR AND ALL OTHER POINTS.
C
      SUBROUTINE EUCLID(DDATA,NUMDAT,IDIM,KCENTR)
C -----------------------------------------------------
      DIMENSION DDATA(1000,10)
      IDIST=IDIM+1
      DO 20 J=1,NUMDAT
      DSQUAR=0.
      DO 10 L=1,IDIM
      DSQUAR=DSQUAR+(DDATA(J,L)-DDATA(KCENTR,L))**2.
   10 CONTINUE
      DDATA(J,IDIST)=DSQUAR
   20 CONTINUE
      RETURN
      END
C
C


C
C THIS SUBROUTINE RESCALES THE DATA AND PSEUDO DATA
C BACK TO ITS ORIGINAL MAGNITUDE.
C
      SUBROUTINE RESCAL(RDATA,NUMBER,IDIM,ZMIN,ZMAX)
C -------------------------------------------------------
      DIMENSION RDATA(1000,10),ZMIN(10),ZMAX(10)
      DO 20 J=1,NUMBER
      DO 10 K=1,IDIM
      RDATA(J,K)=RDATA(J,K)*(ZMAX(K)-ZMIN(K))+ZMIN(K)
   10 CONTINUE
   20 CONTINUE
      RETURN
      END
```

```
C
C
C
C THIS ROUTINE PREPARES FOR THE CALLING OF AN IMSL PLOT ROUTINE SO
C THAT ALL REQUESTED 2-D PLOTS CAN BE MADE.
C
      SUBROUTINE PLOT(PSEUDO,DATA,ZMIN,ZMAX,NPLT,NUMDAT,NUMGEN,NUMPLT)
C ------------------------------------------------------------------
      DIMENSION PSEUDO(1000,10),DATA(1000,10),ZMIN(10),ZMAX(10)
      DIMENSION NUM(8),IHEAD(20),IHEAD1(22),ITITLE(144),JTITLE(144)
      DIMENSION X(1000),Y(1000),IMAG4(5151),ICHAR(10),RANGE(4)
      DIMENSION NPLT(2,40)
C
C ESTABLISH SOME HOLLERITH STRINGS FOR CLARITY OF OUTPUT.
C
      DATA NUM/'1','2','3','4','5','6','7','8'/
      DATA IHEAD/'D','A','T','A',' ','P','O','I','N','T','S',' ',
     +'X',' ',' ','V','S','.',' ','X'/
      DATA IHEAD1/'P','S','E','U','D','O',' ','P','O','I','N','T','S',
     +' ','X',' ',' ','V','S','.',' ','X'/
      DO 10 J=1,144
      ITITLE(J)=' '
      JTITLE(J)=' '
   10 CONTINUE
      DO 20 I=1,20
      L=I+34
      ITITLE(L)=IHEAD(I)
   20 CONTINUE
      DO 30 J=1,22
      L=J+33
      JTITLE(L)=IHEAD1(J)
   30 CONTINUE
      ITITLE(97)='X'
      JTITLE(97)='X'
      ITITLE(126)='X'
      JTITLE(126)='X'
C
C SET VARIABLE VALUES ASKED FOR BY IMSL ROUTINE
C
      M=1
      INC=1
      IOPT=1
C
C BEGIN LOOP WHICH PLOTS FIRST THE DATA AND THEN THE PSEUDO DATA ON TWO
C SEPERATE PLOTS FOR A GIVEN X(I) AND X(J)
C
      DO 60 J=1,NUMPLT
C
C K AND L ARE THE X(Y) AND X(X) COORDINATES RESPECTIVELY
C


      K=NPLT(1,J)
      L=NPLT(2,J)
C
C ESTABLISH THE END POINTS OF THE X AND Y AXES.
C
      RANGE(1)=ZMIN(L)-.10*(ZMAX(L)-ZMIN(L))
      RANGE(2)=ZMAX(L)+.10*(ZMAX(L)-ZMIN(L))
      RANGE(3)=ZMIN(K)-.10*(ZMAX(K)-ZMIN(K))
      RANGE(4)=ZMAX(K)+.10*(ZMAX(K)-ZMIN(K))
C
C ASSIGN THE X AND Y VECTORS VALUES FOR PLOTTING
C
      DO 40 JI=1,NUMDAT
      X(JI)=DATA(JI,L)
      Y(JI)=DATA(JI,K)
   40 CONTINUE
```

```
C
C DOCTOR THE HEADING FOR A DATA PLOT AND SET THE OUTPUT CHARACTER
C
      ITITLE(48)=NUM(K)
      ITITLE(55)=NUM(L)
      ITITLE(98)=NUM(L)
      ITITLE(127)=NUM(K)
      ICHAR(1)='D'
C
C CALL THE IMSL ROUTINE WHICH WILL GIVE BACK A 2-D PLOT
C
      IA=NUMDAT
      CALL USPLT(X,Y,IA,IA,M,INC,ITITLE,RANGE,ICHAR,IOPT,IMAG4,IER)
C
C ASSIGN THE X AND Y VECTORS VALUES FOR PLOTTING
C
      DO 50 J2=1,NUMGEN
      X(J2)=PSEUDO(J2,L)
      Y(J2)=PSEUDO(J2,K)
   50 CONTINUE
C
C DOCTOR THE HEADING FOR A PSEUDODATA PLOT AND SET THE OUTPUT CHARACTER
C
      JTITLE(49)=NUM(K)
      JTITLE(56)=NUM(L)
      JTITLE(98)=NUM(L)
      JTITLE(127)=NUM(K)
      ICHAR(1)='P'
      IA=NUMGEN
C
C CALL THE IMSL ROUTINE WHICH WILL GIVE BACK A 2-D PLOT
C
      CALL USPLT(X,Y,IA,IA,M,INC,JTITLE,RANGE,ICHAR,IOPT,IMAG4,IER)
   60 CONTINUE
      RETURN
      END
```

DISTRIBUTION LIST

No. of Copies	Organization
12	Administrator Defense Technical Info Center ATTN: DTIC-DDA Cameron Station Alexandria, VA 22314
1	Commander US Army Materiel Development and Readiness Command ATTN: DRCDMD-ST 5001 Eisenhower Avenue Alexandria, VA 22333
1	Commander US Army Armament Research and Development Command ATTN: DRDAR-TDC Dover, NJ 07801
2	Commander US Army Armament Research and Development Command ATTN: DRDAR-TSS (2 cys) Dover, NJ 07801
1	Commander US Army Armament Materiel Readiness Command ATTN: DRSAR-LEP-L, Tech Lib Rock Island, IL 61299
1	Director US Army ARRADCOM Benet Weapons Laboratory ATTN: DRDAR-LCB-TL Watervliet, NY 12189
1	Commander US Army Aviation Research and Development Command ATTN: DRDAV-E 4300 Goodfellow Blvd. St. Louis, MO 63120
1	Director US Army Air Mobility Research and Development Laboratory Ames Research Center Moffett Field, CA 94035

No. of Copies	Organization
1	Commander US Army Communications Rsch and Development Command ATTN: DRDCO-PPA-SA Fort Monmouth, NJ 07703
1	Commander US Army Electronics Research and Development Command Technical Support Activity ATTN: DELSD-L Fort Monmouth, NJ 07703
1	Commander US Army Missile Command ATTN: DRSMI-R Redstone Arsenal, AL 35898
1	Commander US Army Missile Command ATTN: DRSMI-YDL Redstone Arsenal, AL 35898
1	Commander US Army Tank Automotive Rsch and Development Command ATTN: DRDTA-UL Warren, MI 48090
1	Director US Army TRADOC Systems Analysis Activity ATTN: ATAA-SL, Tech Lib White Sands Missile Range NM 88002
1	Commander US Army Research Office ATTN: DRXRO-MA, Dr. Robert Launer P.O. Box 12211 Research Triangle Park, NC 27709
1	Commander US Army Combat Development Experimentation Command ATTN: ATEC-SA, Dr. Marion R. Bryson Fort Ord, CA 93941

DISTRIBUTION LIST

Aberdeen Proving Ground

Dir, USAMSAA
    ATTN: DRXSY-D
          DRXSY-MP, H. Cohen
Cdr, USATECOM
    ATTN: DRSTE-TO-F
          DRSTE-AD, Dr. Ted Trybul
Dir, USACSL, Bldg. E3516, EA
    ATTN: DRDAR-CLB-PA

USER EVALUATION OF REPORT

Please take a few minutes to answer the questions below; tear out this sheet, fold as indicated, staple or tape closed, and place in the mail. Your comments will provide us with information for improving future reports.

1. BRL Report Number ______________________________

2. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which report will be used.)

____

____

____

3. How, specifically, is the report being used? (Information source, design data or procedure, management procedure, source of ideas, etc.) ______________________________

____

____

4. Has the information in this report led to any quantitative savings as far as man-hours/contract dollars saved, operating costs avoided, efficiencies achieved, etc.? If so, please elaborate.

____

____

5. General Comments (Indicate what you think should be changed to make this report and future reports of this type more responsive to your needs, more usable, improve readability, etc.) ____________

____

____

____

6. If you would like to be contacted by the personnel who prepared this report to raise specific questions or discuss the topic, please fill in the following information.

Name: ______________________________

Telephone Number: ______________________________

Organization Address: ______________________________

____

____

— — — — — — — — — — FOLD HERE — — — — — — — — — — —

Director
US Army Ballistic Research Laboratory
Aberdeen Proving Ground, MD 21005

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE, $300

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO 12062 WASHINGTON,DC

POSTAGE WILL BE PAID BY DEPARTMENT OF THE ARMY

Director
US Army Ballistic Research Laboratory
ATTN: DRDAR-TSB -S
Aberdeen Proving Ground, MD 21005

— — — — — — — — — FOLD HERE — — — — — — — — — — —